

NAME

CURLMOPT_NOTIFYDATA – custom pointer passed to the notification callback

SYNOPSIS

```
#include <curl/curl.h>
```

```
CURLMcode curl_multi_setopt(CURLM *handle, CURLMOPT_NOTIFYDATA, void *pointer);
```

DESCRIPTION

A data *pointer* to pass to the notification callback set with the *CURLMOPT_NOTIFYFUNCTION(3)* option.

This pointer is not touched by libcurl but is only passed in as the notification callback's **clientp** argument.

DEFAULT

NULL

PROTOCOLS

This functionality affects all supported protocols

EXAMPLE

```
struct priv {
    void *ours;
};

static void notify_cb(CURLM *multi, unsigned int notification,
                     CURL *easy, void *notifyp)
{
    struct priv *p = notifyp;
    printf("my ptr: %p\n", p->ours);
    /* ... */
}

int main(void)
{
    struct priv setup;
    CURLM *multi = curl_multi_init();
    /* ... use socket callback and custom pointer */
    curl_multi_setopt(multi, CURLMOPT_NOTIFYFUNCTION, notify_cb);
    curl_multi_setopt(multi, CURLMOPT_NOTIFYDATA, &setup);
    curl_multi_notify_enable(multi, CURLMNOTIFY_INFO_READ);
}
```

AVAILABILITY

Added in curl 8.17.0

RETURN VALUE

Returns CURLM_OK.

SEE ALSO

CURLMOPT_NOTIFYFUNCTION(3), **curl_multi_notify_disable(3)**, **curl_multi_notify_enable(3)**